

IN THE CLAIMS:

This listing of claims will replace all prior versions, and listings, of claims in the present application:

1. (Original) A method for mirroring data on a first storage server and a second storage server, the method comprising:

queuing write commands at the first storage server between consistency points, the write commands being to write data corresponding to a file system of the first storage server to a local mass storage device coupled to the first storage server;

at a start of each consistency point, sending the write commands to the local mass storage device and to a remote mass storage device coupled to the second storage server;

updating memory blocks of the local and remote mass storage devices based on the write commands; and

at an end of each consistency point constructing a representation to reference each memory block of the local mass storage device that is in use to represent the file system; and sending at least a portion of the representation to the second storage server.

2. (Original) The method of claim 1, wherein the memory blocks of the local and remote mass storage devices comprise address blocks that store a pointer to another block in use by the file system, and data blocks that store data for the file system.

3. (Original) The method of claim 2, wherein each representation constructed at the end of a consistency point is organized as a sub-tree structure comprising nodes that represent one of the address blocks and the data blocks.

4. (Original) The method of claim 3, wherein each sub-tree is rooted in a root node that includes a pointer to each sub-tree structure.

5. (Original) The method of claim 3, wherein sending at least a portion of the representation comprises sending the root node to the second storage server.

6. (Original) The method of claim 5, wherein updating the memory block is in accordance with a scheme in which a memory block referenced within a representation is not overwritten.

7. (Original) The method of claim 6, further comprising allowing read-only access to the remote mass storage device while updating the memory blocks of the remote mass storage server.

8. (Original) A method for operating a first storage server to mirror a file system of the first storage server on a second storage server, the method comprising:

receiving high-level write requests from a client served by the first storage server, the high-level write requests requiring changes to the file system;

generating block-level write commands to update memory blocks of a local mass storage device coupled to the first storage server, based on the high-level write requests;

queuing the block-level write commands; and

at each consistency point, sending the block-level write commands to the local mass storage device and to a remote mass storage device coupled to the second storage server to cause the memory blocks of the local and remote mass storage device to be updated; constructing a representation to reference each memory block of the local mass storage device that is in use to represent the file system after the memory blocks have been updated; and sending at least a portion of the representation to the second storage server.

9. (Original) The method of claim 8, wherein the representation comprises a sub-tree structure in which nodes are used to reference a memory block in use by the file system.

10. (Original) The method of claim 9, wherein the memory blocks comprise address blocks that store a pointer to another block in use by the file system, and data blocks that store data for the file system.

11. (Original) The method of claim 10, wherein each sub-tree structure corresponds to a representation constructed at a particular consistency point is rooted in a root node.

12. (Original) The method of claim 11, wherein sending at least a portion of the representation comprises sending the root node to the second storage server.

13. (Original) A method for operating a second storage server to mirror a file system of a first storage server, the method comprising:

receiving block-level write commands to update memory blocks of a local mass storage device coupled to the second storage server, based on changes to the file system; and

updating the memory blocks in response to receiving the block-level write commands, while allowing read-only access to a backup copy of the file system at the second storage server.

14. (Original) The method of claim 13, wherein the receiving and the updating is performed once at each consistency point.

15. (Original) The method of claim 14, further comprising receiving at the end of each consistency point, a representation of each memory block currently in use to represent an active state of the file system.

16. (Original) The method of claim 14, wherein during updating the memory blocks, memory blocks referenced by the representation are not overwritten.

17. (Original) The method of claim 15, wherein the representation comprises a root node for a tree data structure that includes nodes representing the memory blocks currently in use by the file system.

18. (Original) The method of claim 17, wherein each node in the tree data structure represents one of an address block that stores a pointer to another memory block, and a data block that stores actual data for the file system.

19. (Original) A storage server, comprising:

a processor; and

a memory coupled to the processor, the memory storing instructions which when executed by the processor, cause the storage server to perform a method comprising:

receiving high-level write requests from a client served by the storage server, the high-level write requests requiring changes to a file system of the storage server;

generating block-level write commands to update memory blocks of a local mass storage device coupled to the storage server based on the high-level write requests;

queuing the block-level write commands;

at each consistency point, sending the block-level write commands to the local mass storage device and to a remote mass storage device coupled to another storage server to cause the memory blocks of the local and remote mass storage device to be updated; constructing a representation to reference each memory block of the local mass storage device that is in use to represent the file system, after the memory blocks have been updated; and sending at least a portion of the representation to the other storage server.

20. (Original) The storage server of claim 19, wherein the representation comprises a sub-tree structure in which each node references a memory block in use by the file system.

21. (Original) The storage server of claim 20, wherein the memory blocks comprise address blocks that store a pointer to another block in use by the file system, and data blocks that store data for the file system.

22. (Original) The storage server of claim 21, wherein each sub-tree structure corresponds to a representation constructed at a particular consistency point is rooted in a root node.

23. (Original) The storage server of claim 22, wherein sending at least a portion of the representation comprises sending the root node to the other storage server.

24. (Currently amended) A storage server, comprising:

a processor; and

a memory coupled to the processor, the memory storing instructions which when executed by the processor, cause the processor to perform a method comprising:

receiving block-level write commands to update memory blocks of a local mass storage device coupled to the storage server, based on changes to a file system of ~~the~~ another storage server, the storage server to mirror the file system of said another storage server; and

updating the memory blocks in response to receiving the block-level write commands, while allowing read-only access to a backup copy of the file system at the storage server.

25. (Original) The storage server of claim 24, wherein the receiving and the updating is performed once at each consistency point.

26. (Original) The storage server of claim 25, further comprising receiving at the end of each consistency point, a representation of each memory block currently in use to represent an active state of the file system.

27. (Original) The system of claim 26, wherein during updating the memory blocks, memory blocks referenced by the representation are not overwritten.

28. (Original) The storage server of claim 26, wherein the representation comprises a root node for a tree data structure that includes nodes representing the memory blocks currently in use by the file system.

29. (Original) The storage server of claim 28, wherein each node in the tree data structure represents one of an address block that stores a pointer to another memory block, and a data block that stores actual data for the file system.

30. (Original) A computer-readable medium, having stored thereon a sequence of instructions, which when executed by a computer, cause the computer to perform a method comprising:

receiving high-level write requests from a client served by a first storage server, the high-level write requests requiring changes to a file system of the first storage server;

generating block-level write commands to update memory blocks of a local mass storage device coupled to the first storage server, based on the high-level write requests;

queuing the block-level write commands; and

at each consistency point, sending the block-level write commands to the local mass storage device and to a remote mass storage device coupled to a second storage server to cause the memory blocks of the local and remote mass storage devices to be updated; constructing a representation to reference each memory block of the local mass storage device that is in use to represent the file system, after the memory blocks have been updated; and sending at least a portion of the representation to the second storage server.

31. (Original) The computer-readable medium of claim 30, wherein the representation comprises a sub-tree structure in which each node references a memory block in use by the file system.

32. (Original) A computer-readable medium, having stored thereon a sequence of instructions, which when executed by a computer, cause the computer to perform a method comprising:

receiving block-level write commands to update memory blocks of a local mass storage device coupled to a second storage server, based on changes to a file system of a first storage server; and updating the memory blocks in response to receiving the block-level write



commands, while allowing read-only access to a backup copy of the file system at the second storage server.

33. (Original) The computer-readable medium of claim 32, wherein the receiving and the updating is performed once at each consistency point.

34. (Original) A method, comprising:

in a first storage server, constructing a representation to reference each memory block used to store a file system, the memory blocks being part of a first set of mass storage devices comprising at least one mass storage device coupled locally to the first storage server; and

sending at least a portion of the representation to a second storage server, said portion comprising information to allow reconstruction of the entire representation by the second storage server so that the second storage server has a representation of memory blocks of a second set of mass storage devices comprising at least one mass storage device coupled locally to the second storage server used to store the file system.

35. (Original) The method of claim 34, wherein the method is performed at each of a plurality of consistency points at which a mirror volume contained in the second set of mass storage devices is synchronized with a primary volume contained in the first set of mass storage devices.